Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A decoder ~~adapted to determining a log likelihood logarithmically expressing the probability of passing an arbitrary state on the basis of the value received as soft-input encoded so as to provide at least three or more paths for getting to each state and decoding by using the log likelihood, said decoder~~ comprising:

A2

a path selection means for providing relatively high likelihood of obtaining at least two ~~or more~~ paths of getting to each decoding state from ~~showing a high likelihood out of the~~ at least three ~~or more~~ paths, ~~for getting to each state~~ and for selecting ~~the~~ a maximum likelihood path from ~~the obtained~~ said at least two ~~or more~~ paths,

wherein a log likelihood of getting to a state in the decoder is determined by a soft-input value encoded with a trellis so as to provide at least three paths for getting to the state.

2. (currently amended) The decoder according to claim 1, wherein said path selection means ~~has~~ includes a comparison means for comparing ~~the~~ log likelihoods of all ~~the~~ combinations of said at least two paths selected from ~~all the three or more than~~ said at least three paths for getting to each state.

3. (currently amended) The decoder according to claim ~~1~~ 2, further comprising:

an absolute value selection means for selecting ~~the~~ an absolute value of the difference between ~~the~~ data corresponding to the maximum likelihood path and ~~the~~ data corresponding ~~the~~ a second maximum likelihood path.

4. (currently amended) The decoder according to claim 3, wherein said absolute value selection means ~~has~~ includes an absolute value computing means for computing ~~the~~ absolute values of the difference of ~~each of~~ all ~~the~~ combinations of said at least two paths selected from ~~all the~~ said at least three ~~or more than three~~ paths, ~~getting to each state;~~

wherein the computed absolute values ~~being~~ are compared for magnitude according to ~~on the basis of the information on the~~ outcome of said comparison means~~obtained by comparing the likelihood of each of all the combinations of two paths selected from all the three or more than three paths getting to each state by means of said path selection means~~.

12

5. (currently amended) The decoder according to claim 3, further comprising:

a linear approximation means for computing by linear approximation a correction term added to obtain said log likelihood and expressed by a one-dimensional function relative to a variable,[[;]] said linear approximation means using ~~as~~ said variable ~~the~~ as an absolute value of the difference between ~~the~~ data corresponding to said maximum likelihood path and fed from said absolute value selection means and ~~the~~ data corresponding to said second maximum likelihood path.

6. (currently amended) The decoder according to claim 5, wherein said linear approximation means computes said correction term by expressing ~~the~~ a coefficient representing the gradient of said one-dimensional function for multiplying said variable at least by means of a power exponent of 2.

7. (Original) The decoder according to claim 6, wherein said linear approximation means discards lower bits of an input data according to the power exponent expressing the coefficient representing the gradient of said function.

8. (currently amended) The decoder according to claim 6, wherein said linear approximation means discards ~~the~~ bits from the lowest bit to the k-th lowest bit of the input data when the coefficient representing the gradient of said function is expressed by -2^{-k}.

A 2

9. (currently amended) The decoder according to claim 6, wherein said linear approximation means computes said correction term by expressing the coefficient representing ~~the~~ an intercept of said function by means of a power exponent of 2.

10. (Original) The decoder according to claim 9, wherein said linear approximation means computes said correction term by expressing the coefficient representing the intercept of said function by means $2^{m} - 1$.

11. (currently amended) The decoder according to claim 10, wherein said linear approximation means discards ~~the~~ bits from the lowest bit to the k-th lowest bit of the n-bit input data and inverts ~~the~~ m bits from the k+1-th lowest bit to the m+k-th lowest bit when the coefficient representing the gradient of said function is expressed by means of -2^{-k}.

12. (Original) The decoder according to claim 6, wherein said correction term shows a positive value.

13. (Original) The decoder according to claim 12, wherein said linear approximation means makes the correction term equal to 0 when a negative value is produced by computing said correction term.

14. (currently amended) The decoder according to claim 1, wherein said log likelihood is a log expression of ~~said~~ a probability of getting to a state, using the natural logarithm.

15. (currently amended) The decoder according to claim 1, further comprising:

a first probability computing means for computing for each ~~received~~ soft-input value a first log likelihood logarithmically expressing a first probability determined by a code output pattern and said ~~received~~ soft-input value;

a second probability computing means for computing for each ~~received~~ soft-input value a second log likelihood logarithmically expressing a second probability of getting to each state from a coding starting state in ~~the~~ a time series; and

a third probability computing means for computing for each ~~received~~ soft-input value a third log likelihood logarithmically expressing a third probability of getting to each state from a coding terminating state in ~~the~~ an inverted time series,[[;]]

wherein said second probability computing means and said third probability computing means ~~having~~ includes path selection means~~, respectively~~.

16. (currently amended) The decoder according to claim 15, further comprising:

a soft-output determining means for determining a log soft-output logarithmically expressing ~~the~~ <u>a</u> soft-output in each time slot, using said first log likelihood, said second log likelihood<u>,</u> and said third log likelihood.

17. (currently amended) The decoder according to claim 16, wherein said log soft-output is a <u>natural</u> logarithmic expression of said soft-output ~~with the natural logarithm~~.

A2

18. (currently amended) The decoder according to claim 15, wherein said second probability computing means and said third probability computing means ~~have~~ <u>include</u> absolute value selection means for determining ~~the~~ absolute value<u>s</u> of the difference between ~~the~~ data corresponding to the maximum likelihood path and ~~the~~ data corresponding to the second maximum likelihood path showing ~~the~~ <u>a</u> second highest likelihood~~, respectively~~.

19. (currently amended) The decoder according to claim 15, wherein said second probability computing means and said third probability computing means ~~have~~ <u>include</u> linear approximation means for computing by linear approximation ~~the~~ <u>a</u> correction term added to obtain said log likelihood and expressed by a one-dimensional function relative to a variable~~, respectively~~.

20. (currently amended) The decoder according to claim 1, wherein said log likelihood is determined by computations replacing ~~the~~ multiplications ~~for computing the probability by~~ <u>with</u> logarithmic additions and <u>replacing</u> ~~the~~ additions ~~for computing the probability by~~ <u>with</u> logarithmic maximum value computations.

21. (currently amended) The decoder according to claim 20, wherein a maximum a posteriori probability decoding operation is ~~conducted on the basis of the~~ based on Max-Log-BCJR ~~algorithm~~.

22. (currently amended) The decoder according to claim 5, wherein said log likelihood is determined by computations replacing ~~the~~ multiplications ~~for computing the probability by~~ with logarithmic additions and replacing ~~the~~ additions ~~for computing the probability by~~ with logarithmic maximum value computations and computations of said one-dimensional function.

A2

23. (currently amended) The decoder according to claim 22, wherein a maximum a posteriori probability decoding operation is ~~conducted on the basis of the~~ based on Log-BCJR ~~algorithm~~.

24. (currently amended) The decoder according to claim 1, wherein said path selection means is used to decode convolutional codes ~~are decoded~~.

25. (currently amended) A decoding method ~~adapted to determining a log likelihood logarithmically expressing the probability of passing an arbitrary state on the basis of the value received as a soft-input encoded so as to provide at least three or more paths for getting to each state and decoding by using the log likelihood, said decoding method~~ comprising:

providing relatively high likelihood of ~~a path selection step of~~ obtaining at least two ~~or more~~ paths of getting to each decoding state ~~showing a high likelihood out of the~~ from at least

three ~~or more~~ paths ~~for getting to each state~~, and selecting ~~the~~ a maximum likelihood path from ~~the obtained~~ said at least two ~~or more~~ paths,

wherein a log likelihood of getting to a state in the decoder is determined by a soft-input value encoded with a trellis so as to provide at least three paths for getting to the state.

26. (currently amended) The decoding method according to claim 25, wherein said ~~path selection step~~ providiing includes ~~a comparison step for~~ comparing ~~the~~ likelihoods of all ~~the~~ combinations of said at least two paths selected from ~~all the three or more than~~ said at least three paths ~~getting to each state~~.

27. (currently amended) The decoding method according to claim ~~25~~ 26, further comprising:

~~an absolute value selection step for~~ selecting ~~the~~ an absolute value of the difference between ~~the~~ data corresponding to the maximum likelihood path and ~~the~~ data corresponding ~~the~~ a second maximum likelihood path.

28. (currently amended) The decoding method according to claim 27, wherein said ~~absolute value selection step~~ selecting includes ~~an absolute value computing step for~~ computing ~~the~~ absolute values of the difference of ~~each of~~ all the combinations of said at least two paths selected from ~~all the three or more than~~ said at least three paths. ~~getting to each state;~~

wherein the computed absolute values ~~being~~ are compared for magnitude ~~on the basis of the information on the~~ according to outcome of ~~comparison obtained by comparing the~~

00169352

~~likelihood of each of all the combinations of two paths selected from all the three or more than three paths getting to each state in said path selection step~~ said comparing likelihoods.

29. (currently amended) The decoding method according to claim 27, further comprising:

~~a linear approximation step for~~ computing by linear approximation a correction term added to obtain said log likelihood and expressed by a one-dimensional function relative to a variable[[;]], said linear approximation step using ~~as~~ said variable ~~the~~ as an absolute value of the difference between ~~the~~ data corresponding to said maximum likelihood path and fed in said absolute value selection step and ~~the~~ data corresponding to said second maximum likelihood path.

30. (currently amended) The decoding method according to claim 29, wherein said ~~linear approximation step is adapted to compute~~ computing by linear approximation includes computing said correction term by expressing ~~the~~ a coefficient representing the gradient of said function for multiplying said variable at least by means of a power exponent of 2.

31. (currently amended) The decoding method according to claim 30, wherein said ~~linear approximation step is adapted to~~ computing by linear approximation includes discarding lower bits of an input data according to the power exponent expressing the coefficient representing the gradient of said function.

32. (currently amended) The decoding method according to claim 30, wherein said ~~linear approximation step is adapted to~~ computing by linear approximation includes discarding ~~the~~ bits

from the lowest bit to the k-th lowest bit of the input data when the coefficient representing the gradient of said function is expressed by -2^{-k}.

33. (currently amended) The decoding method according to claim 30, wherein said ~~linear approximation step is adapted to~~ computing by linear approximation includes compute said correction term by expressing the coefficient representing the intercept of said function by means of a power exponent of 2.

A.2

34. (currently amended) The decoding method according to claim 33, wherein said ~~linear approximation step is adapted to~~ computing by linear approximation includes ~~compute~~ computing said correction term by expressing the coefficient representing the intercept of said function by means of $2^m - 1$.

35. (currently amended) The decoding method according to claim 34, wherein said ~~linear approximation step is adapted to~~ computing by linear approximation includes discarding ~~the~~ bits from the lowest bit to the k-th lowest bit of the n-bit input data and to invert the m bits from the k+1-th lowest bit to the m+k-th lowest bit when the coefficient representing the gradient of said function is expressed by means of -2^{-k}.

36. (Original) The decoding method according to claim 30, wherein said correction term shows a positive value.

37. (currently amended) The decoding method according to claim 36, wherein said ~~linear approximation step is adapted to~~ computing by linear approximation includes making ~~make~~ the correction term equal to 0 when a negative value is produced by computing said correction term.

38. (currently amended) The decoding method according to claim 25, wherein said log likelihood is a natural log expression of ~~said~~ a probability of getting to a state ~~with the natural logarithm~~.

A2

39. (currently amended) The decoding method according to claim 25, further comprising:

a first probability computing step for computing for each ~~received~~ soft-input value a first log likelihood logarithmically expressing a first probability determined by a code output pattern and said ~~received~~ soft-input value;

a second probability computing step for computing for each ~~received~~ soft-input value a second log likelihood logarithmically expressing a second probability of getting to each state from ~~the~~ a coding starting state in ~~the~~ a time series; and

a third probability computing step for computing for each ~~received~~ soft-input value a third log likelihood logarithmically expressing a third probability of getting to each state from ~~the~~ a coding terminating state in ~~the~~ an inverted time series,[[;]]

wherein said second probability computing step and said third probability computing step ~~including~~ includes path selection steps ~~respectively~~.

40. (currently amended) The decoding method according to claim 39, further comprising:

a soft-output determining step for determining a log soft-output logarithmically expressing ~~the~~ a soft-output in each time slot, using said first log likelihood, said second log likelihood and said third log likelihood.

41. (currently amended) The decoding method according to claim 40, wherein said log soft-output is a natural logarithmic expression of said soft-output ~~with the natural logarithm~~.

K2

42. (currently amended) The decoding method according to claim 39, wherein said second probability computing step and said third probability computing step include absolute value selection steps for determining ~~the~~ absolute values of the difference between ~~the~~ data corresponding to the maximum likelihood path and ~~the~~ data corresponding to the second maximum likelihood path showing ~~the~~ a second highest likelihood~~, respectively~~.

43. (currently amended) The decoding method according to claim 39, wherein said second probability computing step and said third probability computing step include linear approximation steps for computing by linear approximation ~~the~~ a correction term added to obtain said log likelihood and expressed by a one-dimensional function relative to a variable~~, respectively~~.

44. (currently amended) The decoding method according to claim 25, wherein said log likelihood is determined by computations replacing ~~the~~ multiplications ~~for computing the probability by~~ with logarithmic additions and ~~the~~ replacing additions ~~for computing the probability by~~ with logarithmic maximum value computations.

45. (currently amended) The decoding method according to claim 44, wherein a maximum a posteriori probability decoding operation is ~~conducted on the basis of the~~ based on Max-Log-BCJR ~~algorithm~~.

A2

46. (currently amended) The decoding method according to claim 29, wherein said log likelihood is determined by computations replacing ~~the~~ multiplications ~~for computing the probability by~~ with logarithmic additions and ~~the~~ replacing additions ~~for computing the probability by~~ with logarithmic maximum value computations and computations of said one-dimensional function.

47. (currently amended) The decoding method according to claim 46, wherein a maximum a posteriori probability decoding operation is ~~conducted on the basis of the~~ based on Log-BCJR ~~algorithm~~.

48. (currently amended) The decoding method according to claim 25, wherein said obtaining at least two paths includes decoding convolutional codes ~~are decoded~~.